JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Web Scraping based Product Comparison Model for E-Commerce Website

1st Martina D'Souza
Department of
Information Technology
Xavier Institute of
Engineering, Mumbai
martina.r@xavier.ac.in

2nd Soham Desai
Department of
Information Technology
Xavier Institute of
Engineering, Mumbai
sohamdesai123@gmail.com

3rd Dhruv Agrawal
Department of
Information Technology
Xavier Institute of
Engineering, Mumbai
agrawaldhruv2002@gmail.com

4th Falguni Joshi
Department of
Information Technology
Xavier Institute of
Engineering, Mumbai
joshifalguni0803@gmail.com

Abstract*--In the contemporary era of digital commerce, wherein a myriad of merchandise stands in front of us, the task of discovering the most favorable bargains can be likened to conquering Mount Everest while being blindfolded. This endeavor embarks upon the captivating domain of web scraping, investigating its potential to bring about transformation in traversing the e-commerce landscape. We delve into the methodologies utilized by price comparison websites, dissecting their deployment of web scraping to extract pertinent information about products, compare prices, and equip consumers with invaluable insights. We unravel the secrets behind efficient data extraction, processing, and visualization by employing a technological amalgamation of Beautiful Soup, Flask, Selenium, Node.js, Next.js, and MongoDB. Through this perspective, we unveil the covert mechanisms of electronic commerce platforms, revealing disparities in prices, product variations, and even market trends. This undertaking serves as an illuminating guide, shedding light on the path for consumers to become empowered saviors in online shopping.

***Keywords*–web scraping, e-commerce, price comparison, data analysis, user empowerment, technology stack, Beautiful Soup, Flask, Selenium, Node.js, Next.js, MongoDB**

.

I INTRODUCTION

The rise of the internet has resulted in a radical change in the world of commerce, completely transforming the shopping experience by offering consumers an extensive selection of products that can be easily accessed. Nevertheless, the multitude of choices available often poses a daunting challenge for shoppers seeking to obtain the best possible deals. This is where the concept of web scraping and the emergence of price comparison websites come into play, fundamentally altering the landscape for astute consumers. In our capacity as customers, we navigate through this intricate environment in our pursuit of optimal value, frequently finding ourselves overwhelmed by the deluge of information at our disposal. In this relentless quest, web scraping emerges as a formidable tool, endowing us with the ability to sift through superfluous data and make well-informed purchasing decisions.

The focus of this undertaking centers on exploring the captivating domain of web scraping in the context of e-commerce platforms, elucidating how this technology empowers users to navigate the online marketplace in unprecedented ways. Our analysis will delve into the various methodologies employed by price comparison websites to extract product information, compare prices, and furnish consumers with valuable insights. Through the provision of explicit instances and astute examination, the purpose of this endeavor is to serve as a guiding beacon, leading you through the enthralling world of web scraping for e-commerce websites. This exposition transcends the mere act of price comparison, as we shall venture into the realm of product analysis, whereby we shall delve into the identification of key features, specifications, and even customer reviews, thereby affording a comprehensive view of each product. Our expedition into this domain will take us even deeper, as we explore the transformative process by which the extracted data is converted into actionable intelligence. Picture dynamic comparison tables, interactive visualizations, and personalized recommendations emanating from the raw data illuminate the path toward optimal purchasing decisions.

Through the lens of web scraping, we shall unveil the clandestine workings of e-commerce platforms, laying bare the existence of price discrepancies, variations in product features, and even potential trends within the market.

Our project in e-commerce web scraping employed a potent technological stack to confront the intricacies of extracting data. The role of Beautiful Soup was that of a parsing champion, skillfully dissecting the HTML structures of various online stores and identifying the desirable product information we eagerly sought. As our backend command center, the lightweight web framework Flask adeptly managed data processing, storage, and API interaction. In instances where dynamic elements or complex functionalities emerged, Selenium exhibited its prowess, seamlessly governing web browsers and ensuring seamless data retrieval. Meanwhile, the dynamic duo of Node.js and Next.js collaborated on the front end, transforming our scraped data into user-friendly and interactive interfaces.

Lastly, MongoDB, the reliable NoSQL database, offered a flexible and scalable sanctuary for all the valuable products we uncovered, readily accessible for further analysis and exploration. In summary, this diverse amalgamation of technologies enabled us to navigate the ever-evolving e-commerce landscape and provide valuable insights from the concealed realms of the web.

II BACKGROUND

A. *Web Scrapping*

Web scraping, sometimes known as "web scraping," is a data extraction technique used to automatically collect information from websites. Essentially, it is the process of extracting certain data from web pages using software or scripts. This extracted data can then be saved in a structured format, such as a spreadsheet or database, for further research, visualization, or other uses. Web scraping is sending HTTP queries to web servers, receiving HTML material, and then parsing it to extract the needed data. It's similar to computerised copying and pasting data from a website into a document, but it's done programmatically and on a large scale. While online scraping is not unlawful, it must be done responsibly and lawfully. This includes honoring the website's terms of service. Not overwhelming its servers with requests (which could result in a denial of service), and adhering to data privacy standards while handling personal information. Web scraping can be done manually, however it is typically automated using computer languages like Python, With libraries like Beautiful Soup and Scrapy. These technologies allow you to send queries, parse HTML, and extract data efficiently.

B. *Related* Work

The paper introduces a web application enabling users to compare products on e-commerce sites, utilizing web scraping with Selenium and Python, alongside various algorithms. It demonstrates visually the simulation results, streamlining product evaluation and purchase decisions, and reducing clicks and time. [1] The paper showcases how price comparison websites leverage web scraping to collect and analyze product data from diverse sources, employing PHP libraries to extract details like product names, prices, and ratings. It highlights the benefits for users in making informed decisions and for vendors in marketing their products, proposing a business model centered on free services with revenue from advertising and commissions.[2]

The paper demonstrates the integration of web scraping and machine learning to track and compare product prices across multiple e-commerce platforms, utilizing tools like Beautiful Soup, Scrapy, Java, and Ruby to gather data for analysis. By employing machine learning algorithms, particularly random forest regression, it accurately predicts future prices, enabling users to identify optimal deals and receive email alerts for price drops, ultimately facilitating cost savings and informed purchasing decisions. [3] The paper introduces a method for product price analysis via web scraping, employing crawlers and scrapers to extract data from various e-commerce sites, including product details like price, features, ratings, and reviews. By analyzing and visualizing this data, users can efficiently compare prices and features across multiple platforms, aiding in informed purchasing decisions and saving time and effort. The practical implementation showcased streamlines data extraction reduces labor costs, and offers a unified interface for accessing diverse e-commerce sources, enhancing user convenience and accuracy in price comparison.[4] The system proposed by Dr. G Madhusudhan et al. facilitates remote management of product prices, enabling administrators to schedule data fetching from e-commerce sites, displaying prices on a dashboard, and alerting users when prices meet predefined thresholds. Implemented using .NET and MySQL, it offers real-time tracking of product prices, notifying users when desired price levels are reached, with the flexibility to host on the cloud for broader coverage and use as third-party software for price tracking on other websites.[5]

The paper elucidates on price comparison websites, utilizing web crawling to amass product and pricing data from various online retailers, with web crawlers extracting details like product name, description, price, and availability for comparison. This method aids consumers in finding optimal deals, reshaping the global marketplace dynamics, and bolstering consumer-supplier interactions, showcasing a practical implementation to streamline product comparison, saving users time, effort, and money while providing insights into discounts and offers.[6] The paper by Aditya Ambre et al. underscores the significance of price comparison in e-commerce for cost-conscious consumers, utilizing web crawling and scraping techniques to gather product data from various online retailers, stored in MongoDB. Implemented with Requests and BeautifulSoup4, it optimizes query performance using MongoDB indexing, featuring a system enabling product comparison, filtering, and price tracking via PHP frontend, CRON updates, and wish list functionality, ultimately aiding users in saving time and money with the potential for further expansion.[7]

The research by Pavan Sai Rayalla et al. emphasizes the importance of price comparisons in e-commerce, empowering consumers to make informed decisions and benefiting retailers by increasing visibility and attracting customers. Using web crawling and scraping, Python, BeautifulSoup, and strategies to overcome anti-scraping measures, the paper presents a methodology for data collection, storage, and analysis, demonstrating the impact of price comparison websites on consumer behavior and market trends.[8] The paper by Aswad Shaikh et al. presents a product comparison website leveraging web scraping techniques and a customized algorithm to compare products based on price, features, and user ratings, aiding users in informed purchasing decisions. It showcases the effectiveness of web scraping in data collection and analysis, offering insights for developing similar websites across various domains, integrating user feedback mechanisms, recommendation systems, machine learning, and relevance filtering to enhance functionality and user experience, with continuous improvement through data gathering and user feedback. [9]

The paper by A. Shalini et al. investigates web mining techniques for extracting data from e-commerce websites, focusing on analyzing customer behavior to enhance business strategies and providing users with relevant product comparisons to facilitate informed decision-making and savings. Employing web crawlers and scraping, it builds a system for comparing e-commerce products, enabling users to analyze prices, find the best deals, compare specifications, and access aggregated offers, ultimately saving time and effort while providing accurate information for online shoppers.[10]

The research by Aviraj Jagtap et al. underscores the utility of price comparison websites in consolidating product information from various retailers, facilitating informed decision-making, and saving time and money for users. It highlights the importance of employing web scraping algorithms and libraries like Beautiful Soup or Scrapy to ensure data integrity and legality while emphasizing the empowerment of consumers through comprehensive product details and user reviews.[11] The paper by S. Rajendar et al. introduces Price4You, a price comparison website utilizing Python-based web crawlers and scrapers to gather product details from various e-commerce platforms, facilitating informed purchasing decisions and savings for users. With a user-friendly interface built on the Django framework, Price4You enables price alerts, simplifies the buying process, and aggregates strategies, offers, and deals from leading online stores, ultimately serving as a valuable online shopping assistant for finding the best prices.[12]Kasereka Henrys et al. employ web scraping, utilizing Python's Requests library to gather data efficiently from diverse online sources, particularly beneficial for market analysis in e-commerce, aiding in understanding customer preferences and competitor strategies. The collected data, managed using Python Pandas, facilitates insightful analysis, offering a competitive edge by providing valuable insights into market trends and customer demand.[13]

Faizan Raza Sheikh et al. presents a method for comparing prices across various online shopping sites using web scraping and data analysis, focusing on the importance and methodology of price comparison. Their project utilizes web scraping tools and Python for data collection, cleaning, and analysis, culminating in a user-friendly interface allowing users to input product names and view price comparisons, ultimately aiding in finding the best deals and saving money. [14] Divya Khairkar et al. introduce an e-commerce price tracker employing advanced web scraping methods to collect real-time pricing data from multiple platforms, providing users with personalized notifications for price drops and access to historical price data for informed decision-making. Their project encompasses creating a web scraping tool for active price monitoring, customized alerts, and data analysis, aiming to empower consumers to save money and identify the best online deals. [15]Sandeep Shreekumar et al. delves into web scraping's role in e-commerce, highlighting its significance in gathering competitive intelligence, optimizing pricing strategies, and analyzing customer behavior. They emphasize how web scraping enables market research, enhances data-driven decision-making, and streamlines processes, though noting legal and data quality considerations. Ultimately, web scraping empowers e-commerce enterprises to maintain competitiveness, generate leads, optimize revenue, and stay abreast of competitor activities through efficient data collection and analysis.[16]

K. Varun et al. present a price comparison platform covering various product categories and specifications, addressing limitations while emphasizing features like search, filtering, and prompt response times. Utilizing web crawling, dynamic pricing, and Django, the website aims to provide valuable pricing insights, convenience, and savings for users by comparing prices across multiple websites and promoting transparency in online shopping, facilitated by web scraping and automation for data retrieval.[17] Dilip Kumar Sharma et al. propose a technique for analyzing product specifics on Flipkart and Amazon, leveraging regular website updates to determine the superior platform for each category, facilitated by an administrative interface for configuration and task management. Their methodology involves scraping and preprocessing data, loading it into a database, and utilizing principles-based comparison to forecast optimal procurement platforms, integrating user preferences for personalized product recommendations via UiPath-generated software robots for automated data management and daily updates, ultimately aiming to satisfy end-customer requirements effectively.[18]Shakra Mehak et al. present a website utilizing web scraping and crawling to search product categories across five e-commerce sites, showcasing steps like Python libraries usage, URL retrieval, HTML parsing, and data extraction, achieving 93% accuracy in data retrieval with minimal computation. Their framework dynamically displays favorable deals, foregoing local database storage to enhance storage capacity and processing, ultimately facilitating efficient product searches and comparisons for users. [19]

III IMPLEMENTATION AND METHODOLOGY

In this segment, we've outlined the methodology and execution of our proposed project. Our main goal is to develop a website utilizing web scraping techniques, facilitating users in comparing product prices and quality across various e-commerce platforms. The aim is to streamline the purchasing process, saving users valuable time and effort. Additionally, our approach incorporates Selenium and Chrome web drivers for effective web scraping.

A. Methodology

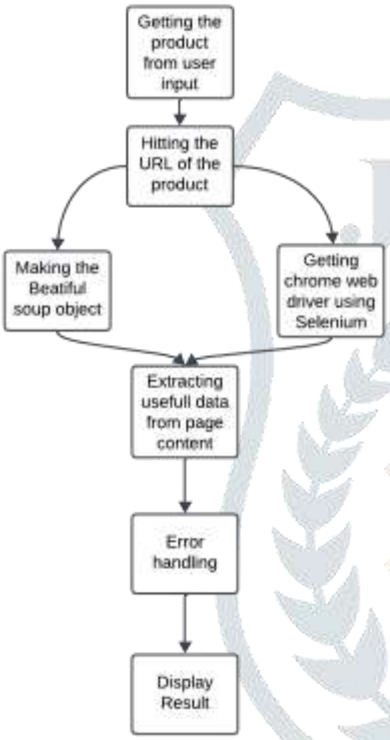
Getting the product from user input
Hitting the URL of the product
Making the Beatiful soup object
Getting chrome web driver using Selenium
Extracting usefull data from page content
Error handling
Display Result

Fig.1- Flow diagram

In the illustration above, the user will enter the product name they wish to compare. Later, our website will construct a URL from several E-commerce websites using the term. Then we produce the beautiful soup object, which contains the web page of the e-commerce website; if the system is unable to fetch the web page using Beautiful Soup, we generate the chrome web drivers using Selenium. Then we extract useful information from the soup object. Then, we handle the mistake that occurs during data extraction from the soup object. Finally, we present the acquired results to the user in a structured style.

B. Implementation

Our proposed application is divided into 7 modules for implementation. Each module is important and should be completed correctly.

1. *Getting the product from user input*

This is the first step in which the users will input the product they want which will be later scrapped from different e-commerce websites.

2. *Hitting the URL of the product*

This stage entails selecting a keyword and then constructing a URL that will be used to direct customers to the e-commerce website. Different URLs are required to reach the appropriate webpage. This can be accomplished by searching for a pattern in the URLs of numerous websites and adding the keyword as a query to each. Creating URLs for each e-commerce website where we want to compare products is the primary goal of this stage. These URLs are of the type that, when entered into our browser's search bar, send us directly to the page on the specific e-commerce website that contains the product to be compared.

3. *Making the beautiful soup object*

The beautiful soup object of the page that was developed in the previous phase must be created in the next step. To use Beautiful Soup, import the Python bs4 module. In the Python terminal you're now using, enter the brief command pip install bs4.

Beautiful Soup is a Python program for parsing HTML and XML documents, making online scraping easier. The approach is streamlined by first parsing the given document into a parse tree that represents its structure. The Soup object that arises from this parse tree conversion serves as the starting point for exploring and extracting data from the document. Beautiful Soup allows users to quickly search for specified components in the parse tree.

4. *Getting Chrome web driver using selenium*

This is an alternative technique to the prior one; in any case, Beautiful Soup will report any errors made when implementing this. The major job of this step is to instruct Chrome's web driver to scrape those objects. The system configuration determines which Chrome web driver is installed. We use Selenium to create these automated web pages. Selenium uses the web drivers found on your machine. Selenium is a powerful web scraping technology that works by automating web browsers. It allows for dynamic user interaction with websites, imitating events such as button clicks, form completion, and page navigation. Selenium requires a web browser (like Chrome or Firefox) that is controlled by the

5. *Extracting useful data from page content*

To perform the web scraping, this step first uses the output from the previous step as its input. This stage makes use of the soup object made up of various web drivers from the previous phase. The primary purpose of this phase is to offer the user with valuable data taken from the website's soup objects, which were already produced in the previous step, allowing them to easily compare the items. The beautiful soup Python module is also utilized to finish this procedure. The data is extracted from the soup object utilising the many wonderful soup functions and properties. Search for each element using its tag name, class name, attribute name, and id. Id and class are the most frequent methods for extracting a specific object from a webpage.

The first step in this article is to identify the class and tag names of the specific element that we wish to target or extract data from. This is easily accomplished by using Google Chrome's developer function. These class and tag names are then used as inputs for the find_all() function, which returns the tag and class names, respectively. The find_all() method generates a list of all matched elements, from which we select the elements with the most important information. Product price, image, URL to the original website, and title are typically regarded as crucial information. To display the image via an HTML IMG element, we utilize ".text" to retrieve the heading from. However, this process may encounter errors, which we discuss in the subsequent step.

6. Error Handling

We may have encountered a type issue in the preceding phase, causing the entire thing to fail and generate an error message. To correct this problem, we should use error handling. To correct this problem, we should use error handling. The find all() function helps us to find the elements. The function discovers all() returns a list of all the elements on the page with the specified class and tag names; however, we do not believe any of these elements are excellent. We require an element that includes all of the product's images, title, and other information.

7. Display the Result to the user

The primary goal of this stage is to provide the user with easy and organized access to the information collected in the previous step, allowing them to easily compare the items using the program's data. Using HTML, CSS, and JS, the Django framework can create an intuitive user interface and show the results in an organized manner.

C. Approach used

During the work, firstly the fetching of the product is done when the user inputs the name of the product in the search box. The name is sent to the flask backend where the functions are defined for scrapping the data from the websites. The HTTP request is sent with the help of the requests module in Python with the search of the product and then when the page is hit, we use the beautiful soup package for scrapping through the web page with the help of HTML tags and CSS which are defined for the name, price, reviews and ratings of the product and the information is fetched

In case there is any error while fetching the information through Beautiful Soup another approach used in backup is the use of Selenium's headless browser functionality which emulates a browser window and helps surf the web pages. With the help of selenium, we get the page source of the web page and then the information is retrieved using the CSS tags. This information is later passed to the front end and displayed to the user.

IV. RESULT

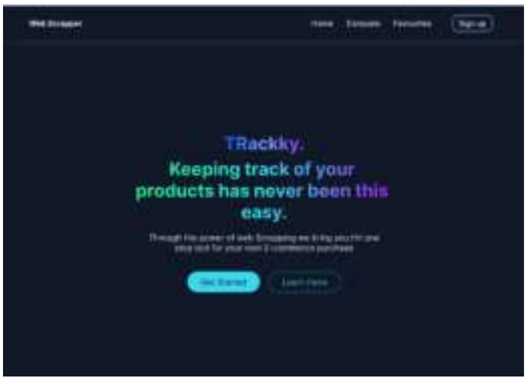

Fig.2- Home page

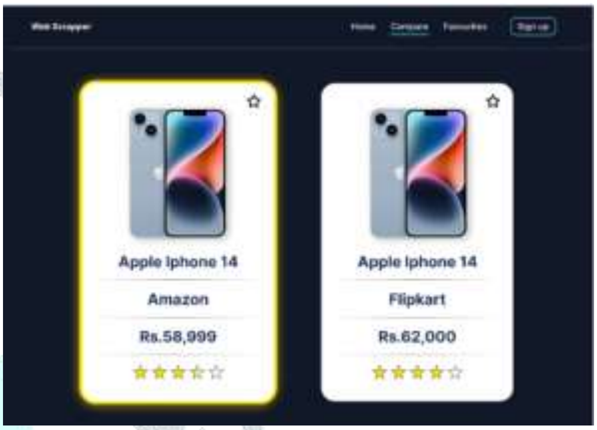

Fig.3- Price Comparison between products

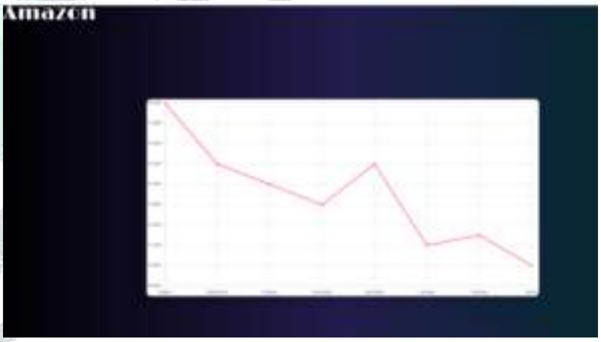

Fig.4- Price Chart for Amazon

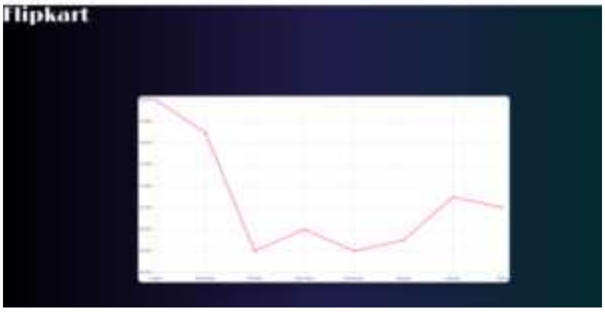

Fig.5- Price Chart for Flipkart

V. CONCLUSION

In conclusion, web scraping serves as more than just a tool to locate the most optimal bargains; it functions as a revolutionary force actively reshaping the e-commerce landscape. By equipping consumers with actionable insights and revealing the concealed mechanisms of online marketplaces, it establishes the groundwork for a shopping experience that is not only more knowledgeable but also more transparent. Our project stands as a testament to the potential for transformation that web scraping possesses, as it effectively demonstrates how this technique can provide individuals with the means to make more intelligent purchasing decisions and enable them to navigate the digital marketplace with astuteness.

VI. ACKNOWLEDGEMENT

The authors would like to acknowledge the guidance and support of Prof. Martina D’Souza during the study and the research work.

VI. REFERENCES

[1]H. Khatter, Dravid, A. Sharma, and A. Kumar Kushwaha, “Web Scraping based Product Comparison Model for E-Commerce Websites,” *IEEE Conference Publication | IEEE Xplore*, Jul. 29, 2022. https://ieeexplore.ieee.org/document/9915892 (accessed Jan. 30, 2024).

[2]A. D. Shaikh, R. Khan, K. Panokher, M. Kr. Ranjan, and V. Sonaje, “E-commerce Price Comparison Website using Web Scraping,” *International journal of innovative research in engineering & multidisciplinary physical sciences*, Jun. 25, 2023. https://doi.org/10.37082/ijirmps.v11.i3.230223

[3]G. V. Kumar, “Web Scraping for E-Commerce Websites,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 10, no. 6, pp. 4114–4119, Jun. 2022, doi: 10.22214/ijraset.2022.44841.

[4]S. B Peedi, Sangeeta , V. Gaded, and Vedhika, “A Novel of Product Price Analysis Using Web Scraping,” *IJIRT*, Aug. 2022, [Online]. Available: https://ijirt.org/master/publishedpaper/IJIRT156159_PAPER.pdf

[5]D. G. Madhusudhan, N. G. Bhat, 2Sahana Venkatraman Patgar, Chandan N A, and Bharath S V, “E-COMMERCE PRODUCT PRICE TRACKER,” *JETIR*, Jun. 2021, [Online]. Available: https://www.jetir.org/papers/JETIR2106691.pdf

[6]M. SOWMIYA, S. CS, M. RAJA M, and S. KUMAR S, “Price Comparison for Products in Various E-Commerce Website,” *IJRTI*, 2023, [Online]. Available: https://www.ijrti.org/papers/IJRTI2305089

[7]A. Ambre, P. Gaikwad, K. Pawar, and V. Patil, “Web and Android Application for Comparison of E-Commerce Products,” *International Journal of Advanced Engineering, Management and Science*, vol. 5, no. 4, pp. 266–268, Jan. 2019, doi: 10.22161/ijaems.5.4.5.

[8]Pavan Sai Rayalla, Setty Vasu Vivek, Sreeja G, Sneha Sanju, and Dr. S. Vijay Kumar, “PRODUCT PRICE COMPARISON ON MULTIPLE E-COMMERCE WEBSITES,” *International Research Journal of Modernization in Engineering Technology and Science*, Jun. 2023, Published, doi: 10.56726/irjmets40965.

[9]A. Shaikh, A. Sonmali, and S. Wakade, “Product Comparison Website using Web scraping and Machine learning.,” *International Research Journal of Engineering and Technology*, Nov. 2023.

[10]A. Shalini and R. Ambikapathy, “E-Commerce Analysis and Product Price Comparison Using Web Mining,” *International Journal of Research Publication and Reviews*, Jun. 2022, [Online]. Available: https://ijrpr.com/uploads/V3ISSUE6/IJRPR5223.pdf

[11]A. Jagtap, “Slickdeals Aggregator Web-Application Using Web Scraping,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 11, pp. 1756–1760, Nov. 2023, doi: 10.22214/ijraset.2023.56913.

[12]S. Rajendar, K. Manikanta, M. Mahendar, and K. Madhavi, “Price Comparison Website for Online Shopping,” *IJCRT*, Jun. 2021, [Online]. Available: https://ijcrt.org/papers/IJCRT2106450.pdf

[13]K. Henrys, “Importance of Web Scraping in E-Commerce and E-Marketing,” *SSRN Electronic Journal*, 2021, Published, doi: 10.2139/ssrn.3769593.

[14]Faizan Raza Sheikh, Hrishabh V. Petkar, Abdul Malik Sheikh, Harsh Lalchand Kose, A. Chandekar, and P. V. Awale, “Price Comparison using Web-scraping and Data Analysis,” *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, Jun. 2023, [Online]. Available: https://ijarsct.co.in/Paper11354.pdf

[15]D. Khairkar, S. Dambhare, O. Gupta, V. Thakre, N. Raut, and Prof. S. N. Dagadkar, “E-Commerce Product Price Tracker using Dynamic Pricing Algorithm,” *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 5, pp. 216–222, May 2023, doi: 10.22214/ijraset.2023.51478.

[16]S. Shreekumar, S. Mundke, and M. Dhanawade, “IMPORTANCE OF WEB SCRAPING IN E- COMMERCE BUSINESS,” *NCRD’s Technical Review : e-Journal*, Jan. 2022, [Online]. Available: https://www.ncrdsims.edu.in/ckfinder/userfiles/files/04(2).pdf

[17]K. Varun, P. Rajesh, P. Dileep, and B.S.V.Satish, “Price Comparison for Online Shopping,” *International Journal of Innovative Science and Research Technology*, Dec. 2023, [Online]. Available: https://www.ijisrt.com/assets/upload/files/IJISRT23DEC258.pdf

[18]D. K. Sharma, S. Lohana, S. Arora, A. Dixit, M. Tiwari, and T. Tiwari, “E-Commerce product comparison portal for classification of customer data based on data mining,” *Materials Today: Proceedings*, vol. 51, pp. 166–171, 2022, doi: 10.1016/j.matpr.2021.05.068.

[19]S. Mehak, R. Zafar, S. Aslam, and S. Masood Bhatti, “Exploiting Filtering approach with Web Scrapping for Smart Online Shopping,” *International Conference on Computing, Mathematics and Engineering Technologies*, 2019, [Online]. Available: https://ieeexplore.ieee.org/document/86733